# AV Board ソフトウェアマニュアル

SIV00-U00

Version 1.0.0-e8b2bcb
2008/07/18

株式会社アットマークテクノ [http://www.atmark-techno.com]

SUZAKU 公式サイト [http://suzaku.atmark-techno.com]

# AV Board ソフトウェアマニュアル

株式会社アットマークテクノ

060-0035 札幌市中央区北 5 条東 2 丁目 AFT ビル 6F
TEL 011-207-6550 FAX 011-207-6570



Version 1.0.0-e8b2bcb
2008/07/18

# 目次

# 図目次

# 表目次

# 1.はじめに

## 1.1. マニュアルについて

このマニュアルは、SUZAKU I/O シリーズの AV ボード(SIV00-U00) に付属している以下のソフトウェアについて記載しております。

- デバイスドライバ
- サンプルアプリケーション

ソフトウェアのカスタマイズなどの開発作業を行う際には、六章に示す参考文献(以降、参考文献と表記)[1][2][3]もあわせてご覧ください。

SUZAKU の機能を最大限に引き出すために、ご活用いただければ幸いです。

## 1.2. フォントについて

このマニュアルでは以下のようにフォントを使っています。

**表 1.1. 使用しているフォント**

| フォント例 | 説明 |
|---|---|
| 本文中のフォント | 本文 |
| [PC ~]$ **ls** | プロンプトとユーザ入力文字列<br>ソースファイルのコード、ファイル名、ディレクトリ名など |

## 1.3. コマンド入力例の表記について

このマニュアルに記載されているコマンドの入力例は、表示されているプロンプトによって、それぞれに対応した実行環境を想定して書かれています。「/」の部分はカレントディレクトリによって異なります。各ユーザのホームディレクトリは「~」で表わします。

**表 1.2. 表示プロンプトと実行環境の関係**

| プロンプト | コマンドの実行環境 |
|---|---|
| [PC /]# | 作業用 PC 上の特権ユーザで実行 |
| [PC /]$ | 作業用 PC 上の一般ユーザで実行 |
| [SUZAKU /]# | SUZAKU 上の特権ユーザで実行 |
| [SUZAKU /]$ | SUZAKU 上の一般ユーザで実行 |

## 1.4. 数字の表記について

このマニュアルに記載されている数字は、特に明記されている場合を除き表記方法によって異なった基数を表します。

**表 1.3. 表示方法と基数の関係**

| 基数 | 記載方法 | 備考 |
|---|---|---|
| 2 進数 | 10100101b | 数字の後に"b"と記載 |
| 10 進数 | 165 | 0 から 9 までのアラビア数字 |
| 16 進数 | 0xA5 | "0x"を数字の前に記載 |

## 1.5. 謝辞

SUZKAU で使用しているソフトウェアは Free Software / Open Source Software で構成されています。Free Software / Open Source Software は世界中の多くの開発者の成果によって成り立っています。この場を借りて感謝の意を示します。

## 1.6. ソフトウェアに関する注意事項

本製品に含まれるソフトウェア (付属のドキュメント等も含みます) は、現状のまま (AS IS) 提供されるものであり、特定の目的に適合することや、その信頼性、正確性を保証するものではありません。また、本製品の使用による結果についてもなんら保証するものではありません。

## 1.7. 保証に関する注意事項

- 製品保証範囲について

  付属品 (ソフトウェアを含みます) を使用し、取扱説明書、各注意事項に基づく正常なご使用に限り有効です。万一正常なご使用のもと製品が故障した場合は、初期不良保証期間内であれば新品交換をさせていただきます。

- 保証対象外になる場合

  次のような場合の故障・損傷は、保証期間内であっても保証対象外になります。

  1. 取扱説明書に記載されている使用方法、または注意に反したお取り扱いによる場合
  2. 改造や部品交換に起因する場合。または正規のものではない機器を接続したことによる場合
  3. お客様のお手元に届いた後の輸送、移動時の落下など、お取り扱いの不備による場合
  4. 火災、地震、水害、落雷、その他の天災、公害や異常電圧による場合
  5. AC アダプタ、専用ケーブルなどの付属品について、同梱のものを使用していない場合
  6. 修理依頼の際に購入時の付属品がすべて揃っていない場合

- 免責事項

弊社に故意または重大な過失があった場合を除き、製品の使用および、故障、修理によって発生するいかなる損害についても、弊社は一切の責任を負わないものとします。



本製品は購入時の初期不良以外の保証をおこなっておりません。保証期間は商品到着後２週間です。本製品をご購入されましたらお手数でも必ず動作確認をおこなってからご使用ください。本製品に対して注意事項を守らずに発生した故障につきましては保証対象外となります。

# 2.作業の前に

## 2.1. 準備するもの

AV ボードを使用する前に、以下のものを準備してください。

- SUZAKU ボード

  40 ピンソケット付き（SUZAKU-S : SZ130-U00 もしくは SUZAKU-V : SZ410-U00）

- 作業用 PC

  Linux もしくは Windows が動作し、1 ポート以上のシリアルポートを持つ PC

- シリアルクロスケーブル

  D-Sub9 ピン（メス - メス）の「クロス接続用」ケーブル

- D-Sub9 ピン-10 ピン変換ケーブル
- AC アダプタ 5V

  DC5V 出力の電源

- ジャンパプラグ 2 個
- NTSC 信号を入出力可能な装置
- 音声 Line を入出力可能な装置
- 4 極ミニプラグ-ピンプラグ×3 ケーブル 2 本 (詳細につきましては参考文献[10]を参照してください)



**注意**

SZ130-U00, SZ410-U00 以外の SUZAKU ボードには対応しておりません。

## 2.2. 接続方法

下の図を参照して接続し、AC アダプタを AV ボードに接続してください。

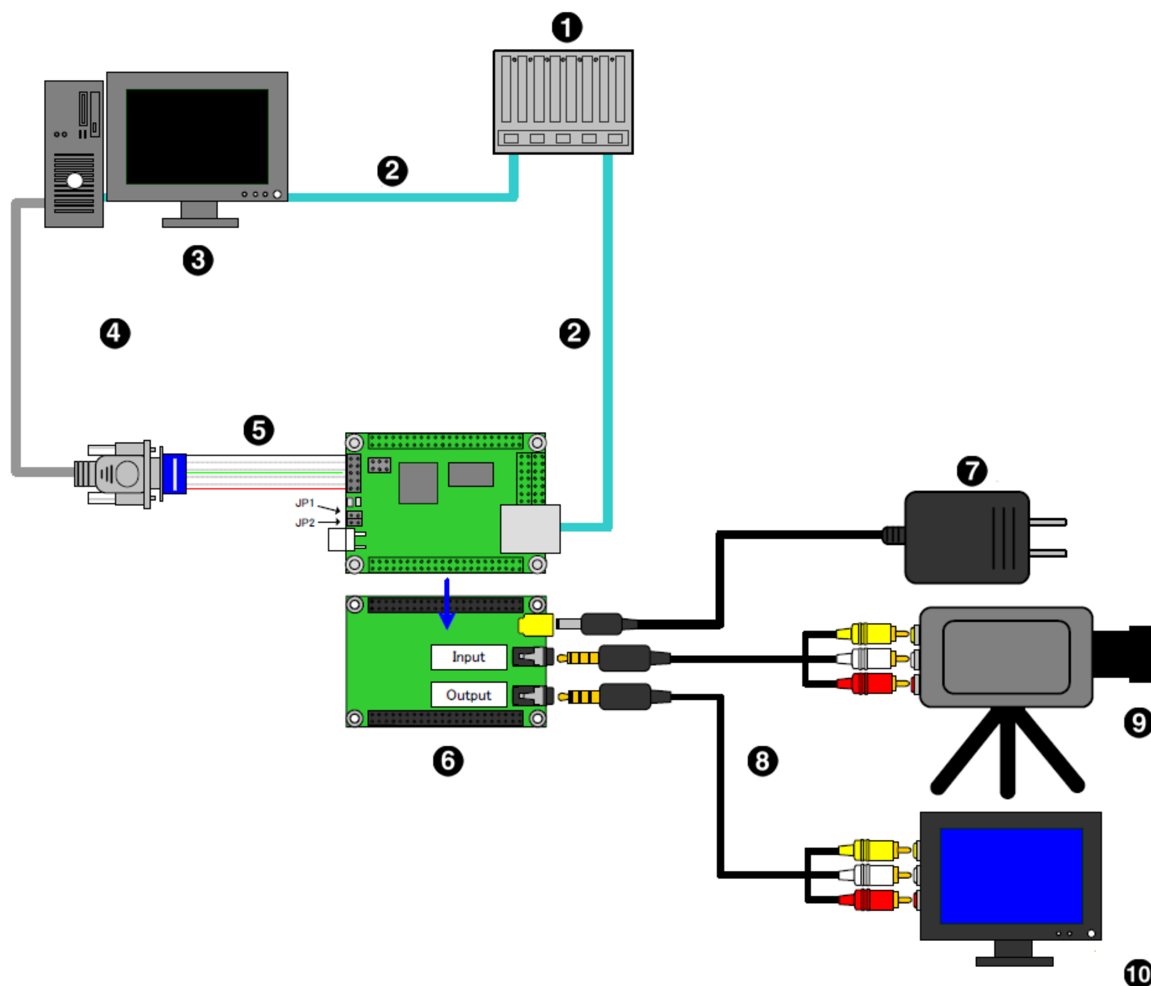



**図 2.1. SIV00-U00 接続例**

❶ HUB
❷ LAN ケーブル
❸ 作業用 PC
❹ D-Sub9 ピンクロスケーブル
❺ D-Sub9 ピン-10 ピン変換ケーブル
❻ SUZAKU+AV ボード
❼ AC アダプタ 5V
❽ 4 極ミニプラグ-RCA ピンプラグ
❾ NTSC、Line 出力が可能な装置
❿ NTSC、Line 入力が可能な装置



**注意**

SUZAKU ボードからは絶対に電源を供給しないでください。電源がショートし、機器を破損する可能性があります。SUZAKU ボードへの電源は、AV ボードから供給されるようになっています。

## 2.3. 開発環境の構築

SUZAKU でのクロス開発環境の構築については、参考文献[3][4]を参照してください。

## 2.4. フラッシュメモリの書き換え

このマニュアルで説明しているアプリケーションおよびデバイスドライバを使用するには、フラッシュメモリ内の FPGA コンフィギュレーションデータおよび Linux イメージを AV ボード用のものに書き換える必要があります。各イメージファイルは、SUZAKU 公式サイトのダウンロードページ [http://suzaku.atmark-techno.com/downloads/all]から取得することができます。

フラッシュイメージの書き換え方法については、参考文献[3][4]を参照してください。



**注意**

FPGA リージョンを不正なデータで書き換えたり、書き換えが異常終了した場合は SUZAKU および AV ボードの電源を入れないでください。最悪の場合、SUZAKU および AV ボードを破壊する恐れがあります。書き換えに失敗した場合は、お使いの SUZAKU 用ハードウェアマニュアルを参照し正しいデータに書き戻してから電源を入れてください。

## 2.5. Linux イメージの作成

Linux イメージを作成するには、make menuconfig 等で AV ボード用のプロダクトを「図 2.2. プロダクトの選択(uClinux-dist)」、「図 2.3. プロダクトの選択(atmark-dist)」のように選択する必要があります。Linux イメージの作成方法については、参考文献[3][4]を参照してください。

```
[PC ~/uClinux-dist]$ make menuconfig
Main Menu
Vendor/Product Selection   --->
--- Select the Vendor you wish to target
(AtmarkTechno) Vendor
--- Select the Product you wish to target
(SUZAKU-S.SZ130-SIV) AtmarkTechno Products
```

**図 2.2. プロダクトの選択(uClinux-dist)**

```
[PC ~/atmark-dist]$ make menuconfig
Main Menu
Vendor/Product Selection   --->
--- Select the Vendor you wish to target
(AtmarkTechno) Vendor
--- Select the Product you wish to target
(SUZAKU-S.SZ410-SIV) AtmarkTechno Products
```

図 2.3. プロダクトの選択(atmark-dist)

# 3.サンプルアプリケーションプログラム

この章では、サンプルアプリケーションプログラムを使用して、フレームバッファデバイスを使用する方法について説明します。プログラムは C 言語で記述されています。またソースコード自体の見通しが良くなるように、機能ごとに特化した作りになっています。

## 3.1. demo-av

### 3.1.1. 概要

demo-av は、Web ブラウザから、AV ボードを操作することのできる Web アプリケーションです。demo-av は以下の操作を行うことができます。

- エフェクト機能を有効にする
- 取得した画像データをブラウザに表示する
- カラーバーを出力する

ソースファイルは以下のディレクトリに収納されています。

**表 3.1. demo-av ソースファイル所在**

| カーネルバージョン | ソースファイル所在 |
|---|---|
| 2.6 系(atmark-dist) | `vendors/AtmarkTechno/SUZAKU-V.Common/siv/demo-av.c` |
| 2.4 系(uClinux-dist) | `user/suzaku/siv/demo-av.c` |

### 3.1.2. 使用法

SUZAKU の起動後、Web ブラウザの URL 欄に http://[SUZAKU の IP アドレス]を指定すると、Web ブラウザに以下のような画面が表示されます。

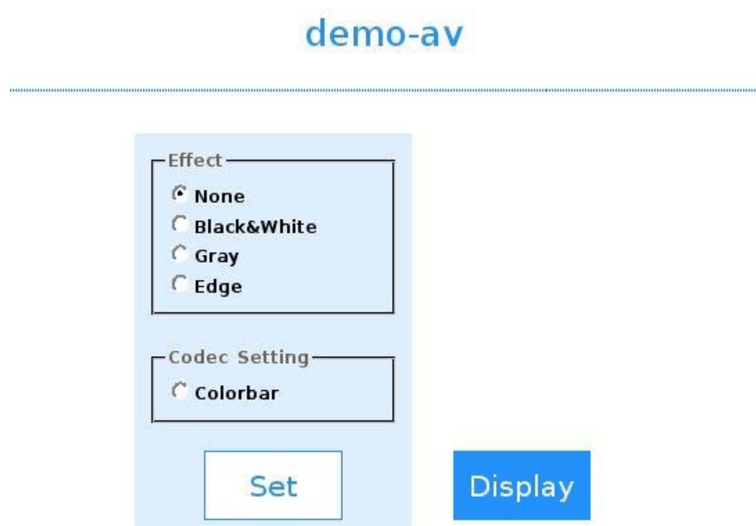



**図 3.1. demo-av 実行画面**

- 取得する画像にエフェクトを加える
  "None", "Black&White", "Gray", "Edge"のいずれかを選択し、「Set」ボタンを押下します。

**表 3.2. エフェクトの説明**

| エフェクト | 説明 |
|---|---|
| None | エフェクトを加えません |
| Black&White | 白黒 (2 値化) |
| Gray | 濃淡 (グレースケール) |
| Edge | エッジ (エッジ検出) |

- Web ブラウザに取得した画像データを表示する
  「Display」ボタンを押下します。取得した画像データを、Windows ビットマップに変換して表示します。画像データの変換から表示するまでの間、画像データの取得が停止します。

- カラーバーの出力
  "Colorbar"を選択し、「Set」ボタンを押下します。カラーバーは、コーデックが直接出力しています。カラーバーを表示している間は「Display」ボタンは表示されません。

# 3.2. demo-impose

## 3.2.1. 概要

demo-impose は、取得した画像に、SUZAKU のロゴをスーパーインポーズして表示するサンプルアプリケーションです。

ソースファイルは以下のディレクトリに収納されています。

**表 3.3. demo-av ソースファイル所在**

| カーネルバージョン | ソースファイル所在 |
|---|---|
| 2.6 系(atmark-dist) | `vendors/AtmarkTechno/SUZAKU-V.Common/siv/demo-impose.c` |
| 2.4 系(uClinux-dist) | `user/suzaku/siv/demo-impose.c` |

## 3.2.2. 使用法

コマンドラインから以下のように入力して実行します。

```
[SUZAKU /]# demo-impose
```

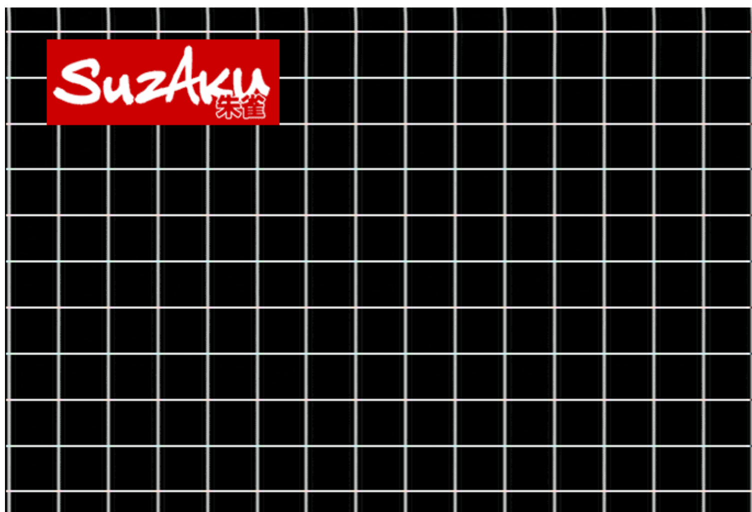



**図 3.2. demo-impose 実行画面**

demo-impose を停止する場合は、Ctrl + c で実行を中断してください。

# 4.サウンドデバイスの使用

この章では、コンソールからサウンドデバイスを使用する方法について説明します。UNIX の標準コマンドや、アプリケーションを使用して、音声データの録音と再生を行います。

## 4.1. 簡単な使用方法

/dev/dsp は、PCM データを扱うことが出来るデバイスファイルです。デフォルトの設定は以下のようになっています。

**表 4.1. デフォルトの設定**

| 項目名 | デフォルト |
|---|---|
| 再生方式 | モノラル |
| サンプリング周波数 | 8kHz |
| データ幅 | 8bit |

事前準備として、読み書き可能なディレクトリ/tmp に移動します。

```
[SUZAKU /]# cd /tmp
```

以下のようにすることで、入力音声をファイルに保存することができます。

```
[SUZAKU /tmp]# cat /dev/dsp > pcm.bin
```

停止する時は Ctrl + c を入力してください。

保存した PCM サウンドファイルは、再生することができます。

```
[SUZAKU /tmp]# cat pcm.bin > /dev/dsp
```

これを応用して、入力した音声を直接再生することも可能です。この場合、実際にはファイル化せず、直接リダイレクションします。

```
[SUZAKU /tmp]# cat /dev/dsp > /dev/dsp
```

## 4.2. アプリケーションの使用

AV ボード用のイメージでは、音声再生アプリケーション bplay と録音アプリケーション brec が用意されています。

ソースファイルは以下のディレクトリに収納されています。

**表 4.2. bplay ソースファイル所在**

| カーネルバージョン | ソースファイル所在 |
|---|---|
| 2.6 系(atmark-dist) | `user/bplay` |
| 2.4 系(uClinux-dist) | `user/bplay` |

brec を使用して、ステレオ/8kHz/16bit として 5 秒分の入力音声を WAV ファイルに保存します。使用しているオプションの意味については「表 4.3. brec の主なオプション」を参照してください。

**表 4.3. brec の主なオプション**

| オプション | 説明 |
|---|---|
| -S | ステレオに設定 |
| -s | サンプリング周波数(Hz) |
| -b | サンプリングのビット幅 |
| -t | 録音時間(sec) |
| -w | WAV フォーマットに設定 |

```
[SUZAKU /tmp]# brec -S -s 8000 -b 16 -t 5 -w tmp.wav
```

32kHz で録音したい場合は、-s に 32000 を指定します。

```
[SUZAKU /tmp]# brec -S -s 32000 -b 16 -t 5 -w tmp.wav
```

以上のように録音したファイルは、PC に転送して Windows のメディアプレーヤーなどで再生可能です。

また、bplay を使用して、WAV ファイルを再生することも可能です。

```
[SUZAKU /tmp]# bplay tmp.wav
```

これを応用して、入力した音声を直接再生することも可能です。

```
[SUZAKU /tmp]# brec -S -s 32000 -b 16 -w | bplay
```

# 5.デバイスドライバ

この章では AV ボードに実装されている各種インターフェースを制御するための Linux 用デバイスドライバについて説明します。

デバイスドライバやデバイスファイルなどは全てデフォルト時の説明となっています。なお、このマニュアルでは Video 用 IP コアと Audio 用 IP コアのアーキテクチャについて、すでに理解されていることを前提としています (参考文献[6][7][8][9]参照) 。

**表 5.1. バージョン**

| 名称 | バージョン | |
|---|---|---|
| | SUZAKU-S | SUZAKU-V |
| Linux Kernel | 2.4.32 | 2.6.18 |
| OPB-SIA00 | 5007100A | - |
| OPB-SIV00 | 5008100A | - |
| XPS-SIA00 | - | 5007100A |
| XPS-SIV00 | - | 5008100A |

## 5.1. ビデオドライバ

### 5.1.1. 概要

**表 5.2. ビデオデバイスドライバ**

| | | | | |
|---|---|---|---|---|
| ドライバ名 | siv | | | |
| ドライバ説明 | SUZAKU I/O AV Board Video Driver | | | |
| デバイスファイル | デバイスファイル名 | タイプ | Major 番号 | Minor 番号 |
| | /dev/fb0 | キャラクタ | 29 | 0 |
| | /dev/fb1 | キャラクタ | 29 | 1 |
| | /dev/fb2 | キャラクタ | 29 | 2 |
| 権限 | 660 (romfs 上にあるため制限あり) | | | |
| ソースファイル所在 | ▼カーネルバージョン 2.6 系(atmark-dist)の場合<br>linux-2.6.x/drivers/video/siv.c<br>linux-2.6.x/include/asm-ppc/suzaku_siv.h<br>▼カーネルバージョン 2.4 系(uClinux-dist)の場合<br>linux-2.4.x/dirvers/vodeo/siv.c<br>linux-2.4.x/include/asm-microblaze/suzaku_siv.h | | | |

なお、本ドライバを使いアプリケーションを開発する際には、コンパイルオプションに-I でカーネルソースの include ディレクトリを追加し、デバイスドライバのヘッダファイルをインクルードしてください。

```
#include <asm/suzaku_siv.h>
```

### 5.1.2. /dev/fb*から使用できるシステムコール

/dev/fb*は、画像データの入出力を行なうためのデバイスファイルです。デフォルトでは「表 5.3. フレームバッファのデフォルト状態」のような状態になっています。

**表 5.3. フレームバッファのデフォルト状態**

| フレームバッファデバイス | 状態 |
| --- | --- |
| /dev/fb0 | 出力 |
| /dev/fb1 | 入力 |
| /dev/fb2 | 未使用 |

open システムコールは、デバイスの使用を開始する際に使用します。

**表 5.4. open システムコール (ビデオドライバ)**

| | |
| --- | --- |
| 書式 | int open(const char *pathname, int flags); |
| 説明 | デバイスをオープンします。オープンに成功した場合、新しいファイルディスクリプタを返します。 |
| 引数 | pathname　オープンするデバイスファイル名<br>flags　ファイルアクセスモード<br>O_RDONLY, O_WRONLY, O_RDWR のどれかひとつを指定します。それぞれ読み込み専用、書き込み専用、読み書き用にファイルをオープンすることを要求します。 |
| 返り値 | 成功した場合は新しいファイルディスクリプタを返し、エラーが発生した場合は-1 を返し、errno の値を変更します。 |

read システムコールは、フレームバッファ内のデータ(32bitRGB 形式) を読み込む際に使用します。

**表 5.5. read システムコール(ビデオドライバ)**

| | |
| --- | --- |
| 書式 | ssize_t read(int fd, void *buf, size_t count); |
| 説明 | デバイスからデータを読み込みます。最大 count バイトをバッファ buf へ読み込みます。 |
| 引数 | fd　ファイルディスクリプタ<br>buf　読み込みデータを格納するバッファ<br>count　読み込みデータのバイト数 |
| 返り値 | 成功した場合は読み込んだバイト数を返し、エラーが発生した場合は-1 を返します。 |

write システムコールは、フレームバッファへデータ(32bitRGB 形式) を書き込む際に使用します。

**表 5.6. write システムコール(ビデオドライバ)**

| | |
| --- | --- |
| 書式 | ssize_t write(int fd, const void *buf, size_t count); |
| 説明 | デバイスへデータを書き込みます。バッファ buf から最大 count バイトをデバイスへ書き込みます。 |
| 引数 | fd　ファイルディスクリプタ<br>buf　書き込みデータを格納するバッファ<br>count　書き込みデータのバイト数 |
| 返り値 | 成功した場合は書き込んだバイト数を返し、エラーが発生した場合は-1 を返します。 |

ioctl システムコールは、ビデオデバイスのさまざまな制御を行う際に使用します。以降に各操作について説明します。

- 画像表示の開始 (FBIO_DISPLAY_ENABLE)
- 画像表示の終了 (FBIO_DISPLAY_DISABLE)
- 画像取得の開始 (FBIO_CAPTURE_ENABLE)
- 画像取得の終了 (FBIO_CAPTURE_DISABLE)
- 画像表示を行うデバイスファイルの設定 (FBIO_DISPLAY_SELECT)
- 画像取得を行うデバイスファイルの設定 (FBIO_CAPTURE_SELECT)
- 画像取得モードの設定 (FBIO_CAPTURE_MODE_SELECT)
- エフェクトの設定 (FBIO_EFFECT_SELECT)
- 画像取得の完了を待つ (FBIO_WAIT_CAPTURE)
- デコーダのレジスタから値を取得 (FBIO_READ_DECODER_REG)
- デコーダのレジスタへ値を設定 (FBIO_WRITE_DECODER_REG)
- エンコーダのレジスタから値を取得 (FBIO_READ_ENCODER_REG)
- エンコーダのレジスタへ値を設定 (FBIO_WRITE_ENCODER_REG)

画像表示の開始 (FBIO_DISPLAY_ENABLE)

**表 5.7. ioctl システムコール (FBIO_DISPLAY_ENABLE)**

| | |
|---|---|
| 書式 | int ioctl(int fd, int cmd); |
| 説明 | 画面表示を開始します。<br>表示するフレームバッファは、FBIO_DISPLAY_SELECT で設定することができます。<br>初期状態では、/dev/fb0 が表示されます。 |
| 引数 | fd　ファイルディスクリプタ<br>cmd　制御コマンド<br>FBIO_DISPLAY_ENABLE を指定します。 |
| 返り値 | 常に 0 を返します。 |

画像表示の終了 (FBIO_DISPLAY_DISABLE)

**表 5.8. ioctl システムコール (FBIO_DISPLAY_DISABLE)**

| | |
|---|---|
| 書式 | int ioctl(int fd, int cmd); |
| 説明 | 画像表示を終了します。終了後は直前に表示していた画像を表示し続けます。 |
| 引数 | fd　ファイルディスクリプタ<br>cmd　制御コマンド<br>FBIO_DISPLAY_DISABLE を指定します。 |
| 返り値 | 常に 0 を返します。 |

画像取得の開始 (FBIO_CAPTURE_ENABLE)

**表 5.9. ioctl システムコール (FBIO_CAPTURE_ENABLE)**

| | |
|---|---|
| 書式 | int ioctl(int fd, int cmd); |
| 説明 | 画面取得を開始します。<br>取得するフレームバッファは、FBIO_CAPTURE_SELECT で設定することができます。<br>初期状態では、/dev/fb1 へ取得します。 |
| 引数 | fd ファイルディスクリプタ<br>cmd 制御コマンド<br>FBIO_CAPTURE_ENABLE を指定します。 |
| 返り値 | 常に 0 を返します。 |

画像取得の終了 (FBIO_CAPTURE_DISABLE)

**表 5.10. ioctl システムコール (FBIO_CAPTURE_DISABLE)**

| | |
|---|---|
| 書式 | int ioctl(int fd, int cmd); |
| 説明 | 画面取得を終了します。 |
| 引数 | fd ファイルディスクリプタ<br>cmd 制御コマンド<br>FBIO_CAPTURE_DISABLE を指定します。 |
| 返り値 | 常に 0 を返します。 |

画像表示を行うデバイスファイルの設定 (FBIO_DISPLAY_SELECT)

**表 5.11. ioctl システムコール(FBIO_DISPLAY_SELECT)**

| | |
|---|---|
| 書式 | int ioctl(int fd, int cmd); |
| 説明 | 画面表示を行うデバイスファイルを設定します。<br>初期状態では、/dev/fb0 が設定されています。 |
| 引数 | fd ファイルディスクリプタ<br>cmd 制御コマンド<br>FBIO_DISPLAY_SELECT を指定します。 |
| 返り値 | 常に 0 を返します。 |

画像取得を行うデバイスファイルの設定 (FBIO_CAPTURE_SELECT)

**表 5.12. ioctl システムコール (FBIO_CAPTURE_SELECT)**

| | |
|---|---|
| 書式 | int ioctl(int fd, int cmd); |
| 説明 | 画面取得を行うデバイスファイルを設定します。<br>画像取得が開始されており、画像取得モードが CAPTURE_STOP_MODE の場合は、画像を 1 つ取得します。<br>初期状態では、`/dev/fb1` が設定されています。 |
| 引数 | fd ファイルディスクリプタ<br>cmd 制御コマンド<br>FBIO_CAPTURE_SELECT を指定します。 |
| 返り値 | 常に 0 を返します。 |

画像取得モードの設定 (FBIO_CAPTURE_MODE_SELECT)

**表 5.13. ioctl システムコール (FBIO_CAPTURE_MODE_SELECT)**

| | |
|---|---|
| 書式 | int ioctl(int fd, int cmd, int mode); |
| 説明 | 画像取得モードを設定します。 |
| 引数 | fd ファイルディスクリプタ<br>cmd 制御コマンド<br>FBIO_CAPTURE_SELECT を指定します。<br>mode CAPTURE_CONTINUOUS_MODE 連続して画像を取得します。<br>CAPTURE_STOP_MODE 画像を 1 つ取得すると画像の取得を停止します。 |
| 返り値 | 成功した場合は 0 を返します。エラーが発生した場合は-1 を返し、errno の値を変更します。 |

エフェクトの設定 (FBIO_EFFECT_SELECT)

**表 5.14. ioctl システムコール (FBIO_EFFECT_SELECT)**

| | |
|---|---|
| 書式 | int ioctl(int fd, int cmd, int effect); |
| 説明 | 取得画像のエフェクトを設定します。<br>デフォルトでは、"なし"が設定されています。 |
| 引数 | fd ファイルディスクリプタ<br>cmd 制御コマンド<br>FBIO_EFFECT_SELECT を指定します<br>effect EFFECT_COLOR なし<br>EFFECT_MONO 白黒<br>EFFECT_GRAY 濃淡<br>EFFECT_EDGE エッジ |
| 返り値 | 成功した場合は 0 を返します。エラーが発生した場合は-1 を返し、errno の値を変更します。 |

画像取得の完了を待つ (FBIO_WAIT_CAPTURE)

**表 5.15. ioctl システムコール (FBIO_WAIT_CAPTURE)**

| | |
|---|---|
| 書式 | int ioctl(int fd, int cmd); |
| 説明 | 初期状態では、`/dev/fb0` が設定されています。 |
| 引数 | fd ファイルディスクリプタ<br>cmd 制御コマンド<br>FBIO_WAIT_CAPTURE を指定します。 |
| 返り値 | 常に 0 を返します。 |

デコーダのレジスタから値を取得 (FBIO_READ_DECODER_REG)

**表 5.16. ioctl システムコール (FBIO_READ_DECODER_REG)**

| | |
|---|---|
| 書式 | int ioctl(int fd, int cmd, struct codec_param *param); |
| 説明 | デコーダのレジスタから値を取得します。<br>引数 param の subaddr メンバで指定したサブアドレスの値を、data に取得します。 |
| 引数 | fd　ファイルディスクリプタ<br>cmd　制御コマンド<br>FBIO_READ_DECODER_REG を指定します。<br>param　subaddr　デコーダのサブアドレス<br>data　取得する値 |
| 返り値 | 成功した場合は 0 を返します。エラーが発生した場合は-1 を返し、errno の値を変更します。 |

デコーダのレジスタへ値を設定 (FBIO_WRITE_DECODER_REG)

**表 5.17. ioctl システムコール (FBIO_WRITE_DECODER_REG)**

| | |
|---|---|
| 書式 | int ioctl(int fd, int cmd, struct codec_param *param); |
| 説明 | デコーダのレジスタへ値を設定します。<br>引数 param の data メンバの値を、subaddr で指定したサブアドレスへ設定します。 |
| 引数 | fd　ファイルディスクリプタ<br>cmd　制御コマンド<br>FBIO_WRITE_DECODER_REG を指定します。<br>param　subaddr　デコーダのサブアドレス<br>data　設定する値 |
| 返り値 | 成功した場合は 0 を返します。エラーが発生した場合は-1 を返し、errno の値を変更します。 |

エンコーダのレジスタから値を取得 (FBIO_READ_ENCODER_REG)

**表 5.18. ioctl システムコール (FBIO_READ_ENCODER_REG)**

| | |
|---|---|
| 書式 | int ioctl(int fd, int cmd, struct codec_param *param); |
| 説明 | エンコーダのレジスタから値を取得します。<br>引数 param の subaddr メンバで指定したサブアドレスの値を、data に取得します。 |
| 引数 | fd　ファイルディスクリプタ<br>cmd　制御コマンド<br>FBIO_READ_ENCODER_REG を指定します。<br>param　subaddr　デコーダのサブアドレス<br>data　取得する値 |
| 返り値 | 成功した場合は 0 を返します。エラーが発生した場合は-1 を返し、errno の値を変更します。 |

エンコーダのレジスタへ値を設定 (FBIO_WRITE_ENCODER_REG)

**表 5.19. ioctl システムコール (FBIO_WRITE_ENCODER_REG)**

| | |
|---|---|
| 書式 | int ioctl(int fd, int cmd, struct codec_param *param); |
| 説明 | エンコーダのレジスタへ値を設定します。<br>引数 param の data メンバの値を、subaddr で指定したサブアドレスへ設定します。 |
| 引数 | fd　ファイルディスクリプタ<br>cmd　制御コマンド<br>FBIO_WRITE_ENCODER_REG を指定します。<br>param　subaddr　エンコーダのサブアドレス<br>data　設定する値 |
| 返り値 | 成功した場合は 0 を返します。エラーが発生した場合は-1 を返し、errno の値を変更します。 |

mmap システムコールは、フレームバッファデバイスをメモリにマップする際に使用します。

**表 5.20. mmap システムコール (ビデオドライバ)**

| | |
|---|---|
| 書式 | int mmap(void *start, size_t length, int prot, int flags, int fd, off_t offset); |
| 説明 | フレームバッファデバイスをメモリにマップします。 |
| 引数 | start　マップするアドレスのヒントを与える<br>length　マップを行うサイズ<br>prot　メモリ保護フラグ<br>PROT_EXEC, PROT_READ, PROT_WRITE, PROT_NONE からひとつ以上、ビット毎に OR 演算したもののいずれかを指定します。<br>flags　マップされたオブジェクトのタイプ、マップ時のオプションやパーミッションを指定します。<br>fd　ファイルディスクリプタ<br>offset　fd のオフセット |
| 返り値 | 成功した場合は 0 を返します。エラーが発生した場合は MAP_FAILED を返し、errno の値を変更します。 |

munmap システムコールは、フレームバッファデバイスをメモリからアンマップする際に使用します。

**表 5.21. munmap システムコール (ビデオドライバ)**

| | |
|---|---|
| 書式 | void *munmap(void *start, size_t length); |
| 説明 | フレームバッファデバイスをメモリからアンマップします。 |
| 引数 | start　アンマップを行うアドレス<br>length　アンマップを行うサイズ |
| 返り値 | 成功した場合は 0 を返します。エラーが発生した場合は MAP_FAILED を返し、errno の値を変更します。 |

close システムコールは、デバイスの使用を終了する際に使用します。

**表 5.22. close システムコール (ビデオドライバ)**

| | |
|---|---|
| 書式 | int close(int fd); |
| 説明 | デバイスをクローズします。 |
| 引数 | fd ファイルディスクリプタ |
| 返り値 | 常に 0 を返します。 |

# 5.2. オーディオドライバ

## 5.2.1. 概要

**表 5.23. オーディオデバイスドライバ**

| | | | | | |
|---|---|---|---|---|---|
| ドライバ名 | sia | | | | |
| ドライバ説明 | SUZAKU I/O AV Board Audio Driver | | | | |
| デバイスファイル | 音声システム | デバイスファイル名 | タイプ | Major 番号 | Minor 番号 |
| | ALSA | /dev/snd/controlC0 | キャラクタ | 116 | 0 |
| | | /dev/snd/pcmC0D0p | キャラクタ | 116 | 16 |
| | | /dev/snd/pcmC0D0c | キャラクタ | 116 | 24 |
| | | /dev/snd/timer | キャラクタ | 116 | 33 |
| | OSS | /dev/mixer | キャラクタ | 14 | 0 |
| | | /dev/dsp | キャラクタ | 14 | 3 |
| | | /dev/audio | キャラクタ | 14 | 4 |
| 権限 | 660 (romfs 上にあるため制限あり) | | | | |
| ソースファイル所在 | ▼カーネルバージョン 2.6 系(atmark-dist)の場合<br>linux-2.6.x/sound/sia.c<br><br>▼カーネルバージョン 2.4 系(uClinux-dist)の場合<br>linux-2.4.x/dirvers/sound/alsa/sia.c | | | | |

本ドライバは Advanced Linux Sound Architecture(ALSA)に準拠して作成されており、また Open Sound System(OSS)互換としても動作するよう設定されています。アプリケーションを開発する際には、ALSA または OSS の各ヘッダファイルをインクルードしてください。
ALSA、OSS についての詳しい情報については、参考文献[11][12][13][14]などを参照してください。

本ドライバは、ALSA 及び OSS として提供される各インターフェースを経由して使用することができます。様々な機能が使用できますが、ここでは OSS として使用できる代表的なシステムコールについて説明します。

## 5.2.2. /dev/dsp から使用できるシステムコール

`/dev/dsp` は、PCM サウンドデータの入出力を行なうためのデバイスファイルです。デフォルトではモノラル/サンプリング周波数 8kHz/データ幅 8bit といったフォーマットの PCM データ向けに設定されます。以下のようなシステムコールが使用できます。

open システムコールは、デバイスの使用を開始する際に使用します。

**表 5.24. open システムコール (オーディオドライバ)**

| | |
|---|---|
| 書式 | int open(const char *pathname, int flags); |
| 説明 | デバイスをオープンします。オープンに成功した場合、新しいファイルディスクリプタを返します。 |
| 引数 | pathname　オープンするデバイスファイル名<br>flags　ファイルアクセスモード<br>録音専用の場合は O_RDONLY を、再生専用の場合は O_RDONLY を、録音再生両方を行なう場合は O_RDWR を指定します。 |
| 返り値 | 成功した場合は新しいファイルディスクリプタを返します。エラーが発生した場合は-1 を返し、errno の値を変更します。 |

ioctl システムコールは、サウンドデバイスのさまざまな制御を行う際に使用します。代表的なコマンドは以下の通りです。

- SNDCTL_DSP_RESET
- SNDCTL_DSP_SYNC
- SNDCTL_DSP_SPEED
- SNDCTL_DSP_STEREO
- SNDCTL_DSP_GETBLKSIZE
- SNDCTL_DSP_SAMPLESIZE

**表 5.25. ioctl システムコール (SNDCTL_DSP_RESET)**

| | |
|---|---|
| 書式 | int ioctl(int fd, int cmd); |
| 説明 | デバイスをリセットします。 |
| 引数 | fd　ファイルディスクリプタ<br>cmd　制御コマンド<br>SNDCTL_DSP_RESET を指定します。 |
| 返り値 | 成功した場合は 0 を返します。エラーが発生した場合は-1 を返し、errno の値を変更します。 |

**表 5.26. ioctl システムコール (SNDCTL_DSP_SYNC)**

| | |
|---|---|
| 書式 | int ioctl(int fd, int cmd); |
| 説明 | 再生の終了を待ちます。 |
| 引数 | fd　ファイルディスクリプタ<br>cmd　制御コマンド<br>SNDCTL_DSP_SYNC を指定します。 |
| 返り値 | 成功した場合は 0 を返します。エラーが発生した場合は-1 を返し、errno の値を変更します。 |

表 5.27. ioctl システムコール (SNDCTL_DSP_SPEED)

| | |
|---|---|
| 書式 | int ioctl(int fd, int cmd, int *arg); |
| 説明 | 入出力サウンドデータのサンプリング周波数を設定します。 |
| 引数 | fd　ファイルディスクリプタ<br>cmd　制御コマンド<br>SNDCTL_DSP_SPEED を指定します。<br>arg　サンプリング周波数を Hz 単位で指定した int 型変数へのポインタを設定します。本ボードが搭載しているコーデックがハードウェア的にサポートしているサンプリング周波数は、8000/32000/48000/96000Hz のいずれかとなっています。 |
| 返り値 | 成功した場合は 0 を返します。エラーが発生した場合は-1 を返し、errno の値を変更します。 |

表 5.28. ioctl システムコール (SNDCTL_DSP_STEREO)

| | |
|---|---|
| 書式 | int ioctl(int fd, int cmd, int *arg); |
| 説明 | 入出力サウンドデータのチャンネル数を設定します。 |
| 引数 | fd　ファイルディスクリプタ<br>cmd　制御コマンド<br>SNDCTL_DSP_STEREO を指定します。<br>arg　モノラルの場合は 0、ステレオの場合は 1 を指定した int 型変数へのポインタを設定します。 |
| 返り値 | 成功した場合は 0 を返します。エラーが発生した場合は-1 を返し、errno の値を変更します。 |

表 5.29. ioctl システムコール (SNDCTL_DSP_GETBLKSIZE)

| | |
|---|---|
| 書式 | int ioctl(int fd, int cmd, int *arg); |
| 説明 | バッファ領域のブロックサイズを取得します。 |
| 引数 | fd　ファイルディスクリプタ<br>cmd　制御コマンド<br>SNDCTL_DSP_GETBLKSIZE を指定します。<br>arg　サイズを受け取る int 型変数へのポインタを設定します。 |
| 返り値 | 成功した場合は 0 を返します。エラーが発生した場合は-1 を返し、errno の値を変更します。 |

表 5.30. ioctl システムコール (SNDCTL_DSP_SAMPLESIZE)

| | |
|---|---|
| 書式 | int ioctl(int fd, int cmd, int *arg); |
| 説明 | 1 サンプルのサイズ(ビット幅)を設定します。 |
| 引数 | fd　ファイルディスクリプタ<br>cmd　制御コマンド<br>SNDCTL_DSP_SAMPLESIZE を指定します。<br>arg　8bit 幅の場合は 8、16bit 幅の場合は 16 を指定した int 型変数へのポインタを設定します。 |
| 返り値 | 成功した場合は 0 を返します。エラーが発生した場合は-1 を返し、errno の値を変更します。 |

read システムコールは、PCM サウンドデータの入力(録音)を行う際に使用します。

**表 5.31. read システムコール (オーディオドライバ)**

| | |
|---|---|
| 書式 | ssize_t read(int fd, void *buf, size_t count); |
| 説明 | デバイスからデータを読み込みます。最大 count バイトをバッファ buf へ読み込みます。 |
| 引数 | fd　ファイルディスクリプタ<br>buf　読み込みデータを格納するバッファ<br>count　読み込みデータのバイト数 |
| 返り値 | 成功した場合は読み込んだバイト数を返します。エラーが発生した場合は-1 を返します。 |

write システムコールは、PCM サウンドデータの出力(再生)を行う際に使用します。

**表 5.32. write システムコール (オーディオドライバ)**

| | |
|---|---|
| 書式 | ssize_t write(int fd, void *buf, size_t count); |
| 説明 | デバイスにデータを書き込みます。最大 count バイトをバッファ buf へ読み込みます。 |
| 引数 | fd　ファイルディスクリプタ<br>buf　読み込みデータを格納するバッファ<br>count　読み込みデータのバイト数 |
| 返り値 | 成功した場合は読み込んだバイト数を返します。エラーが発生した場合は-1 を返します。 |

close システムコールは、デバイスの使用を終了する際に使用します。

**表 5.33. close システムコール (オーディオドライバ)**

| | |
|---|---|
| 書式 | int close(int fd); |
| 説明 | デバイスをクローズします。 |
| 引数 | fd　ファイルディスクリプタ |
| 返り値 | 成功した場合は 0 を返します。エラーが発生した場合は-1 を返します。 |

### 5.2.3. `/dev/mixer` から使用できるシステムコール

`/dev/mixer` は、サウンドデバイスの入出力ミキサーに関する設定を行なうためのデバイスファイルです。以下のようなシステムコールが使用できます。

open システムコールは、ミキサーの使用を開始する際に使用します。

**表 5.34. open システムコール (オーディオミキサードライバ)**

| | |
|---|---|
| 書式 | int open(const char *pathname, int flags); |
| 説明 | ミキサーをオープンします。オープンに成功した場合、新しいファイルディスクリプタを返します。 |
| 引数 | pathname　オープンするデバイスファイル名<br>flags　ファイルアクセスモード<br>O_RDONLY を指定します。 |

| | |
|---|---|
| 返り値 | 成功した場合は新しいファイルディスクリプタを返します。エラーが発生した場合は-1を返し、errno の値を変更します。 |

ioctl システムコールは、入出力ミキサーの設定状態を取得・変更する際に使用します。代表的なコマンドは以下の通りです。

- SOUND_MIXER_READ_ALTPCM
- SOUND_MIXER_WRITE_ALTPCM
- SOUND_MIXER_READ_LINE
- SOUND_MIXER_WRITE_LINE
- SOUND_MIXER_READ_RECSRC
- SOUND_MIXER_WRITE_RECSRC

**表 5.35. ioctl システムコール (SOUND_MIXER_READ_ALTPCM)**

| | |
|---|---|
| 書式 | int ioctl(int fd, int cmd, int *arg); |
| 説明 | ヘッドフォンアウトボリュームを取得します。 |
| 引数 | fd ファイルディスクリプタ<br>cmd 制御コマンド<br>SOUND_MIXER_READ_ALTPCM を指定します。<br>arg ヘッドフォンアウトボリュームを取得する int 型変数へのポインタを設定します。 |
| 返り値 | 成功した場合は 0 を返します。エラーが発生した場合は-1 を返し、errno の値を変更します。 |

ボリューム値は、下位から 8bit(0 〜 7 ビット目)が左チャンネル、その上位 8bit(8 〜 15 ビット目)が右チャンネルとして、各チャンネルの実効値は 0 〜 100 となっています。左右チャンネルとも最大の場合は 25700(16 進数で 0x6464)、左右チャンネルとも最小(ミュート)の場合は 0 となります。以下、3 つのコマンドに関しても同様です。

**表 5.36. ioctl システムコール (SOUND_MIXER_WRITE_ALTPCM)**

| | |
|---|---|
| 書式 | int ioctl(int fd, int cmd, int *arg); |
| 説明 | ヘッドフォンアウトボリュームを設定します。 |
| 引数 | fd ファイルディスクリプタ<br>cmd 制御コマンド<br>SOUND_MIXER_WRITE_ALTPCM を指定します。<br>arg ヘッドフォンアウトボリュームを指定した int 型変数へのポインタを設定します。 |
| 返り値 | 成功した場合は 0 を返します。エラーが発生した場合は-1 を返し、errno の値を変更します。 |
| 備考 | ボリューム値は、100 を指定した場合で+6dB、0 を指定した場合で-73dB(=mute)となっています。不用意に大きな値を設定した場合、接続スピーカー機器を破損する恐れなどがありますので、十分ご注意ください。 |

### 表 5.37. ioctl システムコール (SOUND_MIXER_READ_LINE)

| | |
|---|---|
| 書式 | int ioctl(int fd, int cmd, int *arg); |
| 説明 | ラインインボリュームを取得します。 |
| 引数 | fd ファイルディスクリプタ<br>cmd 制御コマンド<br>SOUND_MIXER_READ_LINE を指定します。<br>arg ラインインボリュームを取得する int 型変数へのポインタを設定します。 |
| 返り値 | 成功した場合は 0 を返します。エラーが発生した場合は-1 を返し、errno の値を変更します。 |

### 表 5.38. ioctl システムコール (SOUND_MIXER_WRITE_LINE)

| | |
|---|---|
| 書式 | int ioctl(int fd, int cmd, int *arg); |
| 説明 | ヘッドフォンアウトボリュームを設定します。 |
| 引数 | fd ファイルディスクリプタ<br>cmd 制御コマンド<br>SOUND_MIXER_WRITE_LINE を指定します。<br>arg ラインインボリュームを指定した int 型変数へのポインタを設定します。 |
| 返り値 | 成功した場合は 0 を返します。エラーが発生した場合は-1 を返し、errno の値を変更します。 |
| 返り値 | ボリューム値は、100 を指定した場合で+12dB、0 を指定した場合で-34.5dB となっています。 |

### 表 5.39. ioctl システムコール (SOUND_MIXER_READ_RECSRC)

| | |
|---|---|
| 書式 | int ioctl(int fd, int cmd, int *arg); |
| 説明 | 録音ソースを取得します。 |
| 引数 | fd ファイルディスクリプタ<br>cmd 制御コマンド<br>SOUND_MIXER_READ_RECSRC を指定します。<br>arg 録音ソースを取得する int 型変数へのポインタを設定します。 |
| 返り値 | 成功した場合はボリューム値を返します。エラーが発生した場合は-1 を返し、errno の値を変更します。 |

録音ソースは、カーネルソースの `include/linux/soundcard.h` にある SOUND_DEVICE_NAMES マクロのインデックスに対応しています。

```
#define SOUND_DEVICE_NAMES    \
        {"vol", "bass", "treble", "synth", "pcm", "speaker", "line", \
         "mic", "cd", "mix", "pcm2", "rec", "igain", "ogain", \
         "line1", "line2", "line3", "dig1", "dig2", "dig3", \
         "phin", "phout", "video", "radio", "monitor"}
```

本ボードは、ラインイン"line"とマイクイン"mic"に対応しています。ioctl で使用する録音ソース番号は、ラインインの場合は"6"、マイクインの場合は"7"に相当します。

次のコマンドに関しても同様です。

### 表 5.40. ioctl システムコール (SOUND_MIXER_WRITE_RECSRC)

| | |
|---|---|
| 書式 | int ioctl(int fd, int cmd, int *arg); |
| 説明 | 録音ソースを設定します。 |
| 引数 | fd　ファイルディスクリプタ<br>cmd　制御コマンド<br>SOUND_MIXER_WRITE_RECSRC を指定します。<br>arg　録音ソースを指定した int 型変数へのポインタを設定します。 |
| 返り値 | 成功した場合は 0 を返します。エラーが発生した場合は-1 を返し、errno の値を変更します。 |

close システムコールは、ミキサーの使用を終了する際に使用します。

### 表 5.41. close システムコール (オーディオミキサードライバ)

| | |
|---|---|
| 書式 | int close(int fd); |
| 説明 | ミキサーをクローズします。 |
| 引数 | fd　ファイルディスクリプタ |
| 返り値 | 成功した場合は 0 を返します。エラーが発生した場合は-1 を返します。 |

# 参考文献


[1] 『*atmark-dist Developers Guide*』』. (株)アットマークテクノ.

[2] 『*uClinux-dist Developers Guide*』. (株)アットマークテクノ.

[3] 『*SUZAKU ソフトウェアマニュアル*』. (株)アットマークテクノ.

[4] 『*SUZAKU スターターキットガイド(FPGA 開発編)*』. (株)アットマークテクノ.

[5] 『*SUZAKU スターターキットガイド(Linux 開発編)*』. (株)アットマークテクノ.

[6] 『*OPB SIA00 Datasheet*』. (株)アットマークテクノ.

[7] 『*OPB SIV00 Datasheet*』. (株)アットマークテクノ.

[8] 『*XPS SIA00 Datasheet*』. (株)アットマークテクノ.

[9] 『*XPS SIV00 Datasheet*』. (株)アットマークテクノ.

[10] 『*AV ボードハードウェアマニュアル*』. (株)アットマークテクノ.

[11] 『*ALSA project - the C library reference*』. http://www.alsa-project.org/alsa-doc/alsa-lib/.

[12] 『*Advanced Linux Sound Architecture (ALSA) project homepage*』. http://www.alsa-project.org/.

[13] 『*Open Sound System Programmer's Guide*』. http://www.opensound.com/pguide/oss.pdf.

[14] 『*Open Sound System Developer Information Site*』. http://developer.opensound.com/.

## 改訂履歴

| バージョン | 年月日 | 改訂内容 |
|---|---|---|
| 1.0.0 | 2008/1/21 | ・ 初版発行 |

AV Board ソフトウェアマニュアル
Version 1.0.0-e8b2bcb
2008/07/18

株式会社アットマークテクノ
060-0035 札幌市中央区北 5 条東 2 丁目 AFT ビル 6F TEL 011-207-6550 FAX 011-207-6570